## AMENDMENTS TO THE CLAIMS

1. (Original) A zero-generating apparatus for use with an instruction set architecture without an r0 register, comprising:

a physical zero register which reads as a zero value;

a Register Alias Table (RAT) for storing an instruction register map; and

a Zeroing Instruction Logic (ZIL) unit for detecting a zeroing instruction and modifying said RAT with a pointer to said physical zero register.

2. (Original) An apparatus in accordance with claim 1, wherein:

said physical zero register is a read only memory (ROM).

3. (Original) An apparatus in accordance with claim 1, wherein:

said ZIL unit detects said zeroing instruction in a trace cache line.

4. (Original) An apparatus in accordance with claim 3, wherein:

an r0 register field logically coupled to said trace cache line for mapping to said physical zero register.

5. (Currently Amended) An apparatus in accordance with claim 3, wherein:

said RAT and said trace cache line are logically coupled to a renaming unit for maintaining said pointer to said physical zero register.

6. (Original) An apparatus in accordance with claim 3, wherein:

said ZIL unit deletes said zeroing instruction from said trace cache line.

7. (Original) An apparatus in accordance with claim 6, wherein:

said ZIL unit modifies a subsequent instruction, where said subsequent instruction is logically coupled to said zeroing instruction within said trace cache line.

8. (Original) An apparatus in accordance with claim 7, wherein:

said ZIL unit modifies said subsequent instruction with an immediate source of zero.

9. (Original) An apparatus in accordance with claim 1, wherein:

said zeroing instruction is an exclusive or (XOR).

10. (Original) An apparatus in accordance with claim 1, wherein:

said zeroing instruction is a subtraction (SUB).

11. (Original) An apparatus in accordance with claim 1, wherein:

said zeroing instruction is a multiply (MUL).

12. (Original) An apparatus in accordance with claim 1, wherein:

said zeroing instruction is a move (MOV).

13. (Original) An apparatus in accordance with claim 7, wherein:

said ZIL unit transforms said subsequent instruction to a MOV instruction.

14. (Original) A zero-generating apparatus for use with a microprocessor, comprising:

a physical zero register which reads as a zero value;

a Zeroing Instruction Logic (ZIL) unit for reading a plurality of instructions and detecting and modifying a zeroing instruction within said plurality of instructions;

where said ZIL unit deletes said zeroing instruction and sets a pointer to said physical zero register in place of said deleted zeroing instruction; and

where said ZIL unit modifies instructions dependent on said deleted zeroing instruction.

15. (Original) An apparatus in accordance with claim 14, wherein:

said ZIL unit modifies instructions dependent on said deleted zeroing instructions with an immediate source of a value when both occur with a single trace cache line.

16. (Original) An apparatus in accordance with claim 14, wherein:

said ZIL unit modifies instructions dependent on said deleted zeroing instructions with a renameable pointer.

17. (Currently Amended) A method of zero-generating with an instruction set architecture without an r0 register, comprising:

detecting a zeroing instruction;

deleting said zeroing instruction;

identifying a subsequent instruction using said zeroing instruction; and

modifying said subsequent instruction with a pointer to a physical zero register which reads as a zero value.

18. (Original) A method in accordance with claim 17, further comprising:

pointing to a physical zero register where said subsequent instruction is not within a common trace cache line.

19. (Original) A method in accordance with claim 17, further comprising:

modifying said subsequent instruction involves replacing instruction sources.

20. (Original) A method in accordance with claim 17, further comprising:

modifying said subsequent instruction involves using a move (MOV) instruction.

21. (Original) A method in accordance with claim 17, further comprising:

said subsequent instruction is modified in response to its location in a trace cache relative to said zeroing instruction.